



Lisa's BASIC-Plus is a flexible programming language that enables you to create new applications for your specific needs. Expanding the BASIC-Plus language used by DEC, BASIC-Plus has a rich set of statements and commands that makes it one of the most versatile BASIC Interpreters available for micro-computers.

A superset of Dartmouth BASIC, BASIC-Plus is simple enough for beginners and powerful enough for developing sophisticated business and scientific applications. And applications written in most other micro-computer BASICs can be easily modified to run on Lisa.™

BASIC-Plus gives programmers one of the largest BASIC programming workspaces available on any personal computer. The IEEE Standard floating-point system provides accuracy that is unsurpassed in other personal or minicomputer BASICs. And as an interpreter, BASIC-Plus is excellent for fast program development and debugging.

Lisa's BASIC-Plus combines the power, precision, and versatility of sophisticated minicomputer BASICs, with the simplicity and ease of use that makes BASIC the preferred language for many business and scientific programming applications.

Uncomplicated programming with fast results.

- Straightforward BASIC language is easy to learn and use.
- Programs can be executed immediately after being entered or changed—no compiling or linking steps are needed.
- Immediate execution mode provides fast results to statements entered.

Compatibility with other BASICs.

- BASIC-Plus is fully compatible with DEC's BASIC-Plus (except for some system-dependent parts).
- Programs written in most microcomputer BASICs can easily be converted to run on Lisa.

Product Highlights

A Powerful, Personal Computer BASIC with Large-System Capabilities

- BASIC-Plus allows all available memory to be used, thus supporting the development of large programs. For easier and more reliable programming, variable names may contain up to 30 significant characters. The IEEE floating point arithmetic provides unsurpassed accuracy in double-precision calculations, and the string arithmetic offers accuracy to 56 digits.

Immediate Response for Fast Results

- For one-time calculations, BASIC-Plus users may enter instructions directly and observe the results as operations are executed. Thus, users can choose to run entire programs or enter statements one at a time, and immediately see the results.

Advanced Features

- BASIC-Plus offers a number of advanced features that enable users to create powerful BASIC programs in an efficient, structured manner. The common IF...THEN statement has been extended with an ELSE clause to simplify conditional IF...THEN...ELSE constructs. And statement modifiers provide an efficient means of writing one-line looping and conditional statements. BASIC-Plus's user-defined functions are extended with multiple lines and parameters to allow far more powerful functions than most BASICs.

Powerful File Handling

- BASIC-Plus supports a variety of file-handling options. Sequential files may be used for reading or writing records in sequential order. Blocks of data may be read or written in any order with the random file I/O. And virtual arrays allow large amounts of data to be stored on disk, while being accessed like ordinary arrays.

Flexible Print Formatting

- For business reporting or file output, the PRINT USING statement provides the most commonly used formatting capabilities, such as control of field length, format of numerical output, leading asterisks, trailing minus signs, floating dollar signs, and punctuation.

Easy Debugging

- Since BASIC-Plus is interpretive, it is easy to debug programs. STOP statements may be inserted into a program at any point; you can then examine and modify the variable values. When you're finished, you can continue program execution from the point it left off. And there is no need to recompile or relink; simply enter the RUN command to start execution.

Workshop Manager

- BASIC-Plus runs in the Workshop, which is a complete development and execution environment with facilities similar to the Apple II and Apple III Pascal environments.

Powerful support for commercial applications.

- PRINT USING option allows flexible report formatting with numbers, character strings, or dollar amounts.
- Double-precision floating point provides accuracy to 15 decimal digits.

Unparalleled support for engineering applications.

- IEEE Standard numerics provide unsurpassed double-precision floating point accuracy.
- String arithmetic offers accuracy to 56 digits.
- Multivariable and multiline functions, recursive functions, virtual arrays, matrix operations, and versatile statements support complex applications.

Efficient development of complex BASIC applications.

- Varied branching and looping statements, statement modifiers, and sequential and random file I/O provide powerful programming facilities.
- Immediate execution mode allows fast and thorough debugging.

Specifications

Operation

The flexible program structure of BASIC-Plus allows both multiple statements per line and multiple lines per statement.

BASIC-Plus also enables users to create and run programs or to enter statements for immediate execution and response (calculator mode):

<i>Program</i>	<i>Immediate Execution</i>
10 A = 3	A = 3
20 PRINT A*A	Ready
30 END	Print A*A
RUN	9
9	Ready
Ready	

Variables

BASIC-Plus offers three types of variables:

- integers (5, 279).
- floating point/scientific notation (.817E55).
- character string ("TO BE OR NOT TO BE").

In addition, one- or two-dimensional arrays may be integer, floating point, or character string. Addition, subtraction, and multiplication may be performed on entire matrices.

Branching and Looping

BASIC-Plus has a powerful set of branching and looping statements:

- GOTO
- IF...THEN...ELSE
- IF...GOTO
- FOR...NEXT
- WHILE...NEXT
- UNTIL...NEXT
- ON...GOTO (computed branch)
- ON...GOSUB (computed subroutine call)
- ON ERROR (branch to error handler)

For example:

- 100 IF BALANCE < 1000 THEN STATUS\$ = 'FINE'
ELSE IF CREDIT\$ = 'OK' THEN STATUS\$ = 'FINE'
ELSE STATUS\$ = 'HOLD'

In addition, BASIC-Plus has statement modifiers for single statement loops and conditional execution:

- IF
- UNLESS
- FOR
- WHILE
- UNTIL

Statements may have single modifiers:

- 100 READ A(I) FOR I = 1 TO 10

Or multiple modifiers:

- 100 READ A(I,J) FOR I = 1 TO 10 IF J < 10

Subroutines

With the GOSUB statement, BASIC-Plus allows sections of code to be executed from various points within a program. Different subroutines may be called from a given point within a program with the ON GOSUB statement.

For example:

- 100 ON I GOSUB 1000, 1100, 1200

The subroutine at line 1000, 1100, or 1200 is executed, depending on whether I = 1, 2, or 3, respectively.

Functions

BASIC-Plus has a comprehensive set of built-in functions for mathematical and string operations. Functions may also be used recursively.

Mathematical functions include:

- Transcendental—sine, cosine, tangent, arc tangent.
- Logarithms and exponents—natural log (base e), common log (base 10), exponent (e^x).
- String arithmetic—add, subtract, multiply, and divide strings of numeric characters (accuracy to 56 places).
- Matrices—transpose, inverse, and determinant.

String functions include:

- Conversions—a full set of conversions between strings and integer, floating point, and ASCII values.
- Position—substrings consisting of the left, right, or middle portion of a string.
- Search—looks for a substring within a string.
- Other—length, translation, and spaces.

For repetitive operations, users may define their own functions, including functions with multiple parameters and more than one line. For example, the following function will find the average of three numbers:

```
10 DEF FNA (A,B,C)
20 SUM = A+B+C
30 FNA = SUM/3
40 FNEND
50 AVERAGE = FNA (2,4,6)
60 PRINT AVERAGE
70 END
RUN
4
Ready
```

I/O

BASIC-Plus offers a comprehensive set of I/O options:

- Console: INPUT statements read from the keyboard; PRINT statements write to the screen.
- Sequential files: enable users to read and write records in sequential order.
- Random (block) files: enable users to read and write files one block at a time in any order.
- Formatted print: PRINT USING statement prints string and numeric values with decimal placement, exponents, leading asterisks, trailing minus signs, floating dollar signs, and commas.
- Arrays: virtual arrays allow users to store arrays on disk rather than in memory, yet be accessed like memory-resident arrays. MAT PRINT prints arrays in a formatted fashion.
- Data lists: READ and DATA statements read data from input data lists and assign the values to specified variables.

Execution Environment

Workshop Manager: a complete development and execution environment for independently developed applications. The workshop facilities are similar to the Apple II and Apple III Pascal environments.

System Requirements

BASIC-Plus runs on any Lisa.

- BASIC-Plus includes:
 - Interpreter.
 - User's Guide.
 - Editor.

BASIC-Plus

**Apple/U.S.**

Apple Computer, Inc.
20525 Mariani Avenue
Cupertino, California 95014
(408) 996-1010
TLX 171-576

Apple/U.K.

Apple Computer (U.K.) Ltd.
Eastman Way
Hemel Hempstead
Herts HP2 7HQ
England
011-44-442-60244
TLX 851-825834

Apple/Europe

Apple Computer International
5/7 rue de Chartres
92200 Neuilly-sur-Seine
France
011-33-1-624-21-13
TLX 842-630296

Apple/Canada

Apple Canada
875 Don Mills Road
Don Mills
Ontario, Canada M3C 1V9
(416) 444-2531
800-268-7637
TLX 06-986561

® Apple and the Apple logo
are registered trademarks of
Apple Computer, Inc.

™ Lisa is a trademark of
Apple Computer, Inc.

Product specifications may
change without notice.